

API - Automation

using Postman with a CSV file

Instructions

You need to copy the pre-request script to your Postman project and modify it according to the following explanation:

Within the pre-request code, there is a variable named **requestBody** in yellow, and inside it, we will write our JSON structure as follows:

```
let requestBody = {  
  MemshakUser: csvData.MemshakUser
```

Enter the original name of the variable in JSON

Fixed term

The name of the field/column in the file that you want to transfer to this variable

pre-request
script

Pre-request

```
let csvData = pm.iterationData.toObject();  
  
for (let key in csvData) {  
  if (csvData[key] === "" || csvData[key] === null) {  
    delete csvData[key];  
  }  
}  
  
let requestBody = {  
  MemshakUser: csvData.MemshakUser,  
  ...  
  ...  
  LstRecordTofes146List: [  
    {  
      MemshakUser: csvData.MemshakUser2,  
      ...  
      ...  
    }  
  ]  
};  
  
function removeEmptyFields(obj) {  
  return JSON.parse(JSON.stringify(obj, (key, value) => value !== undefined ? value :  
    undefined));  
}  
  
requestBody = removeEmptyFields(requestBody);  
  
pm.environment.set("csvData", JSON.stringify(csvData));  
pm.environment.set("requestBody", JSON.stringify(requestBody, null, 2));  
  
console.log(csvData);  
console.log(requestBody);
```

Important note:

It is recommended to name the variables in the CSV file so that they are **identical to the original variable names** in the JSON structure. If there are **identical variable names** in the JSON, whether in the same section of the structure or within any array, a **different number** should be added for each of them.

For example:

This is how a detail from a CSV file looks like:

MemshakUser	Year	Q	IdFinancialTrustee	RowsAmount	MemshakUser2
10	2024	1	205686033	1	11